Zurich Open Repository and Archive
University of Zurich
University Library
Strickhofstrasse 39
CH-8057 Zurich
www.zora.uzh.ch

Year: 2015

The last research mile: achieving both rigor and relevance in information systems research

Nunamaker, Jay F ; Briggs, Robert O ; Derrick, Douglas C ; Schwabe, Gerhard

Abstract: From our desk chairs it may be tempting to work up an idea, build a quick prototype, test it in a lab, and say, "Our work here is done; the rest is merely details." More scholarly knowledge awaits discovery, however, by researchers who shepherd an information systems (IS) solution through the last research mile, that is, through successful transition to the workplace. Going the last research mile means using scientific knowledge and methods to address important unsolved classes of problems for real people with real stakes in the outcomes. The last research mile proceeds in three stages: proof-of-concept research to demonstrate the functional feasibility of a solution; proof-of-value research to investigate whether a solution can create value across a variety of conditions; and proof-of-use research to address complex issues of operational feasibility. The last research mile ends only when practitioners routinely use a solution in the field. We argue that going the last research mile negates the assumption that one must trade off rigor and relevance, showing it to be it a false dilemma. Systems researchers who take their solutions through the last research mile may ultimately have the greatest impact on science and society. We demonstrate the last research mile with cases from our own work and the work of others spanning more than forty years.

DOI: https://doi.org/10.1080/07421222.2015.1094961

Posted at the Zurich Open Repository and Archive, University of Zurich
ZORA URL: https://doi.org/10.5167/uzh-120811
Journal Article
Accepted Version

Originally published at:
Nunamaker, Jay F; Briggs, Robert O; Derrick, Douglas C; Schwabe, Gerhard (2015). The last research mile: achieving both rigor and relevance in information systems research. Journal of Management Information Systems, 32(3):10-47.
DOI: https://doi.org/10.1080/07421222.2015.1094961

The Last Research Mile: Achieving Both Rigor and Relevance in Information Systems Research

Jay F. Nunamaker, Jr
Center for the Management of Information
McClelland Hall
1130 E. Helen Street, Tucson, AZ 85721
jnunamaker@cmi.arizona.edu

Robert O. Briggs (Corresponding Author)
MIS Department, San Diego State University
5500 Campanile Dr. San Diego, CA 92182
rbriggs@mail.sdsu.edu

Douglas Derrick
School of Interdisciplinary Informatics, Peter Kiewit Institute
University of Nebraska at Omaha
1110 S. 67th Street, PKI 172
Omaha, NE 68106
dcderrick@unomaha.edu

Gerhard Schwabe
Department of Informatics
University of Zurich
Binzmuehlestrasse 14, CH-8050 Zurich, Switzerland
schwabe@ifi.uzh.ch

Jay F. Nunamaker, Jr

Jay F. Nunamaker Jr. is Regents and Soldwedel Professor of MIS, Computer Science, and Communication and Director of the Center for the Management of Information and the National Center for Border Security at the University of Arizona. He received his Ph.D. in operations research and systems engineering from Case Institute of Technology, an M.S. and B.S. in engineering from the University of Pittsburgh, and a B.S. from Carnegie Mellon University. He received his professional engineer's license in 1965. Dr. Nunamaker was inducted into the Design Science Hall of Fame in May 2008. He received the LEO Award for Lifetime Achievement from the Association of Information Systems (AIS) in December 2002 and was elected a fellow of the AIS in 2000. He was featured in the July 1997 issue of Forbes magazine on technology as one of the eight key innovators in information technology. He is widely published with an H index of 60. His specialization is in the fields of system analysis and design, collaboration technology, and deception detection. The commercial product, GroupSystems' ThinkTank, based on Dr. Nunamaker's research, is often referred to as the gold standard for structured collaboration systems. He founded the MIS Department at the University of Arizona in 1974 and served as department head for 18 years.

Robert O. Briggs.

As a professor of information systems at San Diego State University, Robert O. Briggs researches the cognitive foundations of collaboration and uses his findings to design new collaborative work practices and technologies. He is cofounder of the collaboration engineering field and co-inventor of the ThinkLets design pattern language for collaborative work processes. He has made exploratory, theoretical, experimental, and technological contributions to the areas of team productivity, technology-supported learning, ideation, creativity, consensus, satisfaction, willingness to change, and technology transition. He has designed collaboration systems and collaborative workspaces for industry, academia, government, and the military. He co-chairs the Collaboration Systems and Technology track for the Hawaii International Conference on System Sciences. He lectures worldwide on collaboration theory and practice, and on the logic of scientific inquiry. He has published more than two hundred scholarly works collaboration systems and technology, addressing issues team productivity, technology-supported learning, creativity, satisfaction, and technology transition. He earned his doctorate Management Information Systems at the University of Arizona in 1994.

Douglas C Derrick

Douglas C. Derrick is an Assistant Professor of IT Innovation, Director of the Applied Innovations Lab, and Co-Director of the Center for Collaboration Science at the University of Nebraska at Omaha. Doug received his PhD in Management Information Systems from the University of Arizona. He holds a Masters degree in Computer Science from Texas A&M University and a Masters degree in Business of Administration from San Jose State University. He is a Distinguished Graduate (top 6%) of the United States Air Force Academy. His research interests include human-agent interactions, intelligent agents, collaboration technologies, decision support systems, persuasive technology and influence. Prior to joining UNO, Dr. Derrick worked as a Program Manager at MacAulay-Brown, Inc. and also served as an

Air Force officer. He has extensive experience working with the Department of Defense. As a contractor and academic, he has been awarded contracts and grants totaling $39.85 Million over the last 12 years (principle investigator awards total $16.89 Million). Doug has published 41 peer-reviewed journal articles and conference proceedings.

Gerhard Schwabe

Gerhard Schwabe is a full professor of at the University Zurich in the Department of Informatics. He researches the intersection of collaborative technologies and information management. Since starting as researcher in 1988, he studied collaboration at the granularity of dyads, small teams, large teams, organizations, communities and social networks. In doing so, he follows either an engineering approach ("design science") or an exploratory approach - frequently in collaboration with companies and public organizations. He graduated from the Technical University of Darmstadt in 1988 and received his doctoral and postdoctoral education at the University of Hohenheim. He was appointed tenured professor at the University of Koblenz-Landau in 1998, and full professor at University of Zurich in 2002. In Zurich he leads the information management research group. He has published in computer science conferences (like the ACM conference on CSCW & Social Computing) and well as in major information systems conferences and journals.

The Last Research Mile: Achieving Both Rigor and Relevance in Information Systems Research

Abstract

From our desk chairs it may be tempting to work up an idea, build a quick prototype, test it in a lab, and say, "Our work here is done; the rest is merely details." More scholarly knowledge awaits discovery, however, by researchers who shepherd an IS solution through the last research mile, that is, through successful transition to the workplace. Going the *last research mile* means using scientific knowledge and methods to address important unsolved classes of problems for real people with real stakes in the outcomes. The last research mile proceeds in three stages: *Proof-of-concept* research to demonstrate the functional feasibility of a solution; *proof-of-value* research to investigate whether a solution can create value across a variety of conditions; and *proof-of-use* research to address complex issues of operational feasibility. The last research mile ends only when practitioners routinely use a solution in the field. We argue that going the last research mile negates the assumption that one must trade off rigor and relevance, showing it to be it a false dilemma. Systems researchers who take their solutions through the last research mile may ultimately have the greatest impact on science and society. We demonstrate the last research mile with cases from our own work and the work of others spanning more than 40 years.

Beyond Desk-chair Science

On any given day, a senior academic in a systems discipline could outline a 25-year research agenda over lunch. There is consequently not sufficient time in one career to research all that one can imagine. From our desk chairs it may therefore be tempting to work up an idea, build a quick prototype, test it in a lab, and say, "Our work here is done; the rest is merely details." We can only make modest progress, however, with our desk chair inventions. The defining purpose of Information Systems as an academic discipline is *to understand and improve the ways people create value with information* [56]. Inspired inventions are necessary, but not sufficient to achieve that purpose. Individuals ascribe value along a number of dimensions (Table 1).

[TABLE 1 ABOUT HERE]

Perceptions of value have both a magnitude and a direction: positive or negative. To create positive value, therefore, system solutions must be not only technically and economically feasible, but also *operationally feasible*, which, for purposes of this paper, we define as politically, socially, cognitively, emotionally, and physically acceptable to its stakeholders [1]. Vastly more scholarly knowledge therefore awaits discovery by researchers who shepherd a solution through the *last research mile*, that is, through successful transition to the workplace.

Given the complexity of operational feasibility, the nature of a useful system solution may not be intuitive or obvious from the desk chair. The complexities of operational feasibility can only be

[1] We choose the term, *acceptable*, carefully in this definition. Accepting a thing is not the same as liking a thing. Information systems solutions, for example, are likely to empower some stakeholders while disempowering others (political value). While those who lose power may not like it, so long as they use the solution – even with reservations or under protest – and do not thwart its use by others, they have, by definition, accepted it.

discovered, understood, and solved in the field. These discoveries and understandings become valuable nuggets of scholarly knowledge that may inform decades of subsequent research and practice. Those nuggets are found in the details, and for systems research, the details are found in the field, and the field is the domain of the last research mile for systems research. Further, the true societal impacts that give research its relevance can only manifest in practice. A researcher who travels the last mile can therefore leave behind a legacy of fully rigorous scholarly contributions that are simultaneously directly relevant to the workplace and society. We will argue, therefore, that going the last research mile therefore negates the assumption in systems fields that researchers must trade off rigor and relevance.

Consider an early low-tech solution where unanticipated nuggets of scholarly knowledge emerged from operational feasibility issues, and informed follow-on research. In 1953 Alex Osborn proposed a brainstorming protocol meant to increase team creativity [58]. Participants were to propose ideas under four constraints: a) don't screen out bad ideas; b) welcome unusual ideas; c) combine and improve ideas; d) withhold criticism. A moderator captured ideas on a public list. Brainstorming had intuitive appeal from the desk chair, but 20 subsequent scientific studies could not demonstrate improved creativity [26]. Two constructs interfered with brainstorming productivity:

a. P*roduction blocking*, people either forgot their own ideas while listening to others, or ignored others to remember their own ideas.
b. *Evaluation apprehension*, people did not contribute ideas unpopular ideas for fear of negative consequences.

These nuggets informed early research on electronic brainstorming systems, where participants could contribute simultaneously, thereby eliminating production blocking and could contribute ideas anonymously, thereby eliminating evaluation apprehension [23, 30]. This knowledge would not have been gained had research ended with the publication of the initial solution conceived from the desk chair.

In a high-tech case, one of us (Schwabe) and his graduate students worked with a bank in Switzerland to improve a system for financial advisors to lead new clients through a financial planning process [57]. The system displayed a sequence of screens to capture details of a client's financial situation (e.g. current investments, risk preferences, and cash flows). From the desk chair, it met its functional requirements. Advisors, however, said the system interfered with the advisory relationship because they had to turn their backs on their clients to use the keyboard and screen, and so could not make eye contact as they used it. The system also presented a fixed sequence of screens, one for each aspect of the advisory process, but the sequence conflicted with the non-linear way clients reasoned about their finances. Further, participants working on later steps could not see relevant details recorded in earlier steps. They could not, for instance, see current investments while considering risk preferences. Further still, the original screens were too small for shared viewing. The bank therefore switched to larger screens, but that left clients feeling their privacy had been compromised. Clients tended not to understand the process, not to understand the financial information, and not to trust the advisor. Clients and advisors reported low satisfaction with the system.

Researchers implemented a new system on a large, horizontal tabletop touch screen computer around which advisors and clients could sit and make eye contact (Figure 1) [38].

[FIGURE 1 ABOUT HERE]

They refactored the interview sequence tool into independently accessible widgets that shared data, but could be used in parallel and in any order. Users could rotate widgets with their fingers and pass them to other participants as one passes a paper across a table, which gave clients and advisors co-equal control of the process (Figure 2).

[FIGURE 2 ABOUT HERE

Interestingly, the advisory process improved on all measures when the advisor and client sat at adjacent sides of the table rather than opposite sides. The adjacency yielded two distinct work spaces: a) a relationship-building space where users made eye contact and established shared understanding; and b) an artifact work space where users exercised joint control and of the screen objects (Figure 1). With privacy, process flow, transparency, and usability issues sorted out, users better understood the advisory process and the financial information, and deemed the advisors to be more trustworthy. Clients and users reported deeper satisfaction with the process and outcomes of advisory sessions. [38, 57]. None of these insights were obvious when the solution was conceived from the desk chair. Beyond the technical contribution, the research produced scholarly insights about process and information transparency and relationship building in the context of advisory interactions that would not have been realized had research ended with the first implementation.

The Last Research Mile in Three Stages

The *last research mile* involves using scientific insights and methods to address important unsolved classes of problems for real people with real stakes in the outcomes. The last mile begins with *proof-of-concept* research to demonstrate the functional feasibility of a potential solution. In the context of this essay, *functional feasibility* means the degree to which a potential solution is technically possible, and the degree to which it is within the mental and physical abilities of its intended users or participants.

The last research mile continues with *proof-of-value* research, the second stage where researchers investigate whether stakeholders can use the solution to create value across a variety of contexts and conditions.

The third stage, *proof-of-use* research seeks to create self-sustaining and growing communities of practice around a generalizable solution, and to demonstrate that practitioners can successfully create and gain value from their own instances of the generalizable solution. Each stage is a rich source of new, relevant research questions because new, unanticipated phenomena related to functional and operational feasibility tend surface.

Each stage of the last research mile has different goals, and so produces prototype solutions with different properties. Each stage has the potential to make different kinds of contributions to scholarly knowledge supported by a variety of research approaches. Similar benefits, however, accrue to the researchers who pursue each stage (See Table 2 for a summary). This section elaborates the details of proof-of-concept, proof-of-value, and proof-of-use research.

Proof-of-Concept Research

The key goals for the proof-of-concept stage are:

a. To demonstrate the functional feasibility for a potential solution to an important class of unsolved problems in the field.
b. To develop deeper and broader understandings of the class of problems addressed by a solution.
c. To discover the first nuggets of scholarly knowledge that may lead to future operational feasibility for a solution.

d. To begin research on scholarly theories that explain outcomes-of-interest so as to better-inform design choices as research progresses.

The definitive research products for this stage are, accordingly, proof-of-concept prototypes. A proof-of-concept prototype is usually a rudimentary solution that is not necessarily scalable, not necessarily full-featured, and not necessarily stable or optimized, but will have sufficient functionality to test functional feasibility with simple tasks. A proof-of-concept prototype would not usually be useful to support real work in the field. Proof-of-concept prototypes tend to be substantially less expensive than those produced for proof-of-value and proof-of-use research. In the spirit of creative destruction, they are quick-and-dirty, meant to be tried and thrown away as they engender new knowledge that suggests better design choices.

Proof-of-concept research, however, can produce contributions beyond working prototypes. It can contribute, for example, rigorous exploratory research to identify phenomena-of-interest, their correlates, and the contexts and conditions where they manifest [66]. Among the most important phenomena-of-interest for proof-of-concept research are:

a. Unacceptable outcomes that define the problem space
b. Constructs that appear to correlate with those outcomes
c. Functional and operational constraints on the solution space

These insights provide useful clues that may, in turn, inform subsequent theoretical research, and follow-on experimental research to test those theories. Early research in collaboration technology, for example, showed that sometimes the most productive users were the least satisfied [21]. People who felt dissatisfied with their experience, even for reasons having nothing to do with technology would nonetheless decline to use the system [7], even in the face of compelling evidence that they derived substantial financial benefits from the system. Existing theories of satisfaction – e.g. attribute theories, goal attainment theories, and disconfirmation theories – could not explain the full range of observed phenomena. This led to more than a decade of theory development and testing to explain satisfaction phenomena, culminating with Yield Shift Theory, a causal model to predict and explain the onset, direction, and magnitude of satisfaction responses [18].

Proof-of-concept studies tend to use simple tasks, course-grained treatments and course-grained measurements of relationships between use-of-the-system-as-a-whole and outcomes of interest that a solution might address, with the objective to gain an overall sense of whether an approach is promising. Discovering the phenomena-of-interest and their correlates in the problem space also gives a researcher deeper understandings of the goals of and barriers faced by problem stakeholders, which, in turn, informs the design goals and indicators of quality for useful solutions. Such efforts establish a tight connection between research rigor and relevance. Exploratory research is validated through concatenation [66]. Concatenation in exploratory research plays a role similar to that played by replication in experimental research. When one exploratory researcher reports discovery of a particular relationship among constructs-of-interest in a particular context under particular conditions, then others concatenate that work by exploring how that relationship is similar or different under other conditions and in other contexts. Concatenating studies continue as scholarly contributions up to the point of conceptual saturation, which is to say, when no new variations in the relationships-of-interest can be discovered. In experimental research, however, replications that produce different results are deemed to be refutations of the original work, because the purpose of an experiment is to test the logic of a causal theory. In exploratory research, by contrast concatenations that produce different results are deemed to increase our understanding the phenomenon of interest, because the purpose of exploratory research is to discover and describe phenomena, their correlates, and the conditions and contexts under which they manifest.

Proof-of-concept prototypes have a higher risk of failure than do later prototypes: a class of problems sometimes turn out to be insignificant; the search for phenomenon-of-interest may be elusive; early solutions may not work. Speed and exploratory rigor are therefore useful to produce rich understandings of how and why people respond to various early approaches. The failure of a prototype, however, does not constitute a failure of research. The problems and opportunities in the stakeholder domain, the private and organizational goals of the stakeholders, the economic, political, social and operational constraints in the environment, and, perhaps, accounts of prior dead-ends can and should be reported as exploratory findings [66] or as case studies [71], so long as the work is conducted under the standards of rigor for those research methods. The insights gained through failure often lead to better theory and more-sophisticated prototypes. Quantitative experimental rigor is less useful in proof-of-concept research. Proof-of-concept research, however, quickly lays a foundation for the next stage of the last research mile.

Proof-of-concept researchers can gain advantages over-and-above those that accrue from publishing papers that say, in effect, “The prototype works!” Before they develop a work system solution [2], for example, researchers must assure themselves that an important class of unsolved problems exists in the field. If they use the disciplines of exploratory research to conduct this work, they can publish one or more landmark papers that set the research agenda for a new branch of scholarly inquiry. Having defined the class of problems, they must define and publish generalizable requirements for solutions to the class of problems. If also conducted under the disciplines of applied science/engineering research [19], this work can also yield landmark papers that inform the development of alternative solutions. Furthermore, a working prototype is compelling evidence to grant sponsors and other key stakeholders that a potential solution is possible, and that the researcher has a credible approach. The roughest demonstration is therefore more compelling than the most polished proposal. A sponsor who sees a poor-but-functioning proof-of-concept prototype, along with detailed explanations for how it could be improved has much more reason to believe the desired outcome is possible than a sponsor who reads a proposal for a radical new solution that has never been attempted. Indeed, sometimes flaws in the prototype are, themselves persuasive. For example, two of us (Nunamaker and Briggs) showed a balky proof-of-concept prototype for a collaborative decision platform to a potential sponsor with a high-stakes need for a solution. We were uncertain whether the prototype would survive the demonstration, but it held together until the very end of the demonstration, at which point it had a fatal crash and could not be restarted. We thought the sponsor would decline to fund further work. Instead, he said, “I wasn’t going to fund you because it looked like you had already solved the problem. When it crashed, though, I realized you were nearer to the beginning than the end. I’ll fund the project.”

Sometimes, when researchers develop a proof-of-concept prototype, they assume that value-creation and use-in-the-workplace will follow as a matter of course. Proof-of-concept research, however, is rarely sufficient to yield the deeper insights necessary to design economically, technically, and operationally feasible implementations of new work systems. The potential impact of the research is not yet realized. While useful contributions have been made, it is usually necessary for the researcher to move on to more-sophisticated proof-of-value research in order to maximize the value created by a research Proof-of-Value Research

Proof-of-Value research seeks to demonstrate that a solution can be useful for real problems. The primary goals for Proof-of-Value research are:

a. To deepen scientific understandings of the phenomena discovered during Proof-of-value research, and to discover and describe new phenomena pertinent to the problem and its potential solutions.
b. To measure the degree to which a generalizable solution meets its design goals for improving key outcomes
c. To improve the functional quality of the solution. The *functional quality* means the quality of both the technological components of a solution and of the processes by which the technology can be used to create value
d. To discover and describe unintended consequences of a solution
e. To develop and document the processes by which and the conditions under which a solution can be used to create value.
f. To understand better the technical, economic, and operational feasibility factors that might affect successful deployment of such a solution in the workplace.

One key objective for proof-of-value research is to separate utility from superstition and supposition; to learn which elements of the solution have what effects on the outcomes of interest; to discover and explain the independent and interaction effects of various design choices; to distinguish the solution elements that create positive value from those with neutral or negative value; to gain knowledge about ways the technical components of a work systems solution can be used in the field to create value.

Epistemological objectives of this stage are:

a. To gain deeper theoretical understanding of the causal mechanism that may underpin the phenomena-of-interest – the outcomes the solution seeks to improve
b. To clarify the meanings that stakeholders ascribe to their experiences of the solution
c. To analyze the degree to which the solution might relieve, or the risk that the solution might create systemic social injustice for some group of people.

Qualitative and quantitative observations of field trials with real tasks, formal theory development, and quantitative experimental rigor are all therefore common features for proof-of-value research as are qualitative interpretivist inquiry and criticalist ethical research.

A technical objective of this stage is to develop design principles for the potential solution; to gain deeper understandings of the principles of form and function pertaining to the solution and how they relate to achieving design goals [33]. Proof-of-value researchers take their prototypes into the lab for rigorous exploration and experimentation, and out to the field so people with a real stake in the outcome can try to use the solution for real work. Proof-of-value prototypes must therefore be sufficiently full featured and robust to solve a least one important task in the field, and must have sufficient stability, functionality, and performance to survive field trials that may last weeks or months. Non-functional requirements and integration into a natural work environment may still be achieved through improvisation, workarounds, and extra technical and operational support.

A practical objective Proof-of-value research is to gain insights on both the expected and serendipitous benefits the stakeholders can achieve with the new solution, as well as the unanticipated and unintended negative outcomes that may emerge. Proof-of-value research often reveals new economic, political, social, and other operational constraints that block users from deriving value from a demonstrably valuable solution.

Proof-of-value researchers develop bodies of explicit knowledge augmented by rich bodies of tacit knowledge about the problem space the solution space. This helps them predict how seemingly minor, sometimes counterintuitive design choices could have powerful effects on the efficacy of a solution. When preparing grant proposals, this knowledge gives them distinct advantages over those who have not traversed the last research mile. Where others can only sketch the general outlines of a problem and propose vague blue-sky solutions, a proof-of-value researcher can be specific and detailed about the parameters of the problem space, approaches to a solution, and the reasons why the proposed solution should be useful.

Their rich bodies of knowledge and experience also give proof-of-value researchers an advantage in producing scholarly publications. Field experience reveals relevant phenomena and gives opportunities to study them with exploratory rigor [66]. If the literature does not fully explain the phenomena, then the last mile researcher's expertise provides added insights toward development of new causal theory [4]. When design choices for a proof-of-value prototype are informed by theoretical logic, then the last mile researcher can simultaneously test a theory and a solution with both experimental rigor and operational relevance.

Despite the critical importance of proof-of-value research to the development of new work system solutions, even an extensive, robust, multi-university proof-of-value research stream may not be sufficient to produce solutions that are ready for the work place. Systems in the field are integral to the rich, dynamic complexity of organizational operations, and so must not only be functionally effective, but functionally feasible, which, as noted above, encompasses political, social, cognitive, emotional, and physical interests of the stakeholders. Information is power, for example, so apart from the economic and cognitive value a new IS solution may create with respect to its design goals, it will also have political implications for success-critical stakeholders.

It is rarely possible to anticipate from the desk chair how the elements of operational feasibility may play out for a new work system solution. One of us (Briggs) worked with a large military command, for example, to design a new solution for a high-stakes process that had a nine-month operational cycle. We conducted proof-of-concept and proof-of-value research on a solution that could have cut the cycle to 3 days. It is axiomatic military doctrine that the army that can think and act the fastest can prevail against a physically superior adversary by forcing the adversary to respond to conditions that no longer exist. From the desk chair, therefore, we reasoned that the new solution would be immediately and warmly adopted. Mission accomplished.

Instead, the two-star general in charge of the process declined the new solution, saying, "This is clearly the right answer, but we absolutely cannot use it." His reason: the secretaries didn't like it. The new solution disempowered them with respect to outcomes they valued. Under the old solution, they were responsible for controlling the flow of certain kinds of information throughout the command. It was a mission-critical task, they performed it well, and doing so gave them power and influence in the organization. With the new system, they would have no responsibility, no control, and no influence. Speaking with candor, the general said, "My success as a general depends completely on the secretaries. If they don't like the solution, they can easily ruin my career, and if we implement this solution, they will." We had to reconceive the solution to address an functional feasibility issue we could not have anticipated from the desk chair.

Moreover, as noted by Luna-Reyes, Derrick, Langhals and Nunamaker [47], policy and technology development cannot be developed independently. Policy development and technical development processes should be integrated to ensure selected policy and technologies are compatible and implemented in an effective way that meets the stated goals. Policy decisions should not rush under-developed and unproven technology to the field prematurely, and systems should meet stated policy goals.

Proof-of-use Research

The primary goals for the proof-of-use stage of the last research mile is:

a. To determine whether it is possible to create self-sustaining and growing communities of practice around a new technical solution.
b. To codify a design theory encapsulating the knowledge practitioners require to develop successfully their own instances of a generalizable solution [33].
c. To continue deepening scholarly understandings of the problem and solution spaces.

One key objective of this stage, therefore, is to discover, describe, understand, and design to accommodate functional feasibility issues - political, social, cognitive, emotional, and physical - that would otherwise prevent people from deriving value from a new IS solution. The depth of knowledge that emerges from this research sometimes leads to a complete reimagining of the solution in ways that produce significantly more value than did the proof-of-value systems.

Both exploratory and experimental rigor may be useful in this stage. Because proof-of-use research specifically targets improving outcomes in the work place, it can often be conducted in with support from partner organizations or spin-offs from the research institution. The rigor is thereby married to relevance and impact on a large scale.

Proof-of-use research frequently involves one or more rounds of complete redesign and reimplementation as new insights are discovered, documented, and tested. It is common in proof-of-use research to discover that the technology-components of a proof-of-value prototype were hard-coded to assumptions that don't hold in the workplace. One of us (Briggs) worked on solution, for example, to support collaborative agile software development. The solution yielded strong proof-of-value results. The system, however, was with a hard-coded assumption that each project had a single leader who had sole access to certain privileges. That assumption held well for co-located teams on the same work schedule. When left in the field for proof-of-use studies, however, it was quickly deployed to support follow-the-sun development. When the project leader went to bed in one time zone, people in the other time zones could no longer use the system effectively. The whole architecture of the system had to be changed to support multiple project leaders.

By definition, the rich tacit knowledge a researcher gains from the proof-of-value stage is not readily transferrable to others. As a consequence, it is sometimes the case that a demonstrably valuable solution is not immediately useful to practitioners who do not have the benefit of that knowledge. Another objective for proof-of-use research is therefore to externalize and codify tacit knowledge. This often leads to new scholarly insights that, in turn, inform ways to design and deploy a solution so that it can be useful to practitioners. It also informs more-generalizable theory and broader understandings in the field.

Proof-of-use solutions must be sufficiently full-featured and robust that they can be left behind as part of the users' everyday work environment without ongoing support from the researcher. The increasing numbers of users in this stage leads to an increasing need for improvements to non-functional requirements such as stability, scalability and maintainability. Integrating a proof-of-use solution into the workplace may require technical integration, data integration as and integration of new technologies into organizational work processes. Such an initiative may reveal additional concerns and constraints that did not emerge during proof-of-value research. Another objective for proof-of-use research is therefore to discover, understand, address, and document those concerns for practitioners who may instantiate their own instances of the solution.

As researchers work to establish proof-of-use, they continue to discover new nuggets of knowledge. They gain deep understandings of the technical, operational, and economic aspects of the problem domain and the solution space. At this stage, the gems and nuggets of scholarly knowledge tend to unify into sophisticated, integrated understandings, cumulating in a design theory – a collection of requisite knowledge that would allow practitioners successfully to create their own instances of the generalizable solution. Design theories include, for example, concepts and theories, principles of form and function, structured design methodologies, and expository instances of the solution [33]. A last-mile research will have expository instances of the solution by definition, and by creating and fielding such instances, may gain the knowledge required to complete a design theory.

Last mile researchers can often create value intentionally for problem owners, and can even discover new nuggets of knowledge intentionally, because they now know where to look. As work continues, a Proof-of-use research may give rise a commercial products and services. Interestingly, proof-of-use research often continues after commercialization as new issues emerge in the marketplace. Commercialization initiatives, in turn, often generate new resources to further advance the research.

Last Mile Research in Action

The last research mile is rarely a straight, smooth road. It is more often a winding road that doubles back on itself, forks, and forks again. It leads through rich scientific country, however, and can yield not only useful systems solutions, but also exciting exploratory, theoretical, experimental, and applied science/engineering contributions. Consider, for example, a stream of research from our own work that began with proof-of-concept work on PSL/PSA in the 1960's, which spawned, in turn, research streams on Computer-aided Software Engineering (CASE), Group Support Systems (GSS), Collaboration Engineering (CE), and Credibility Assessment Technologies.

The Last Research Mile for PSL/PSA.

In the early days of software, developers struggled to verify the completeness, consistency, and correctness of requirements. To address that challenge, one of us (Nunamaker) worked with Daniel Teichroew and others to develop the Problem Statement Language (PSL)/ Problem Statement Analyzer (PSA) as a potential solution (Teichroew and Hershey III 1977). PSL was a structured English syntax for statements representing information systems requirements. PSA was a program that could analyze PSL to discover inconsistent, incomplete, or incorrect requirements. For proof-of-concept, the research team used PSL to document its own requirements for PSA, then built a PSA prototype to meet those requirements. They ran the PSL documents through the prototype to see if it would work. It did. Proof-of-concept was achieved.

The research team discovered in the proof-of-concept phase, however, that people outside the research team found it difficult to learn how to write their requirements in the original PSL syntax. They therefore redesigned the syntax through numerous iterations to make it easier for non-engineers to use (Figure 3).

[FIGURE 3 ABOUT HERE]

The proof-of-concept results gave the team sufficient credibility to gain additional funding to develop the system further, and take it into the field for Proof-of-Value trials. In 1972, the PSL/PSA team received a grant from the U.S. Navy to help gather requirements for a new inventory management system from 4000 stakeholders (Nunamaker et al, 1976). It was, at the time, one of the largest software projects ever undertaken, and was a daunting challenge. PSL/PSA seemed a promising approach because, it was reasoned, users could write their requirements in the structured English of PSL, and PSA would validate them.

In the field, however, most stakeholders refused to write their requirements in PSL, even in its new, more user-friendly form. They expressed displeasure with the format and the style of reasoning the approach required in statements like, "I can't do this! I'm not a programmer!" The general population did not wish to think in terms of algorithms and data models. Still facing the challenge of integrating requirements from so many stakeholders, the Navy hired consultants from Haskins and Sells to sit with stakeholders, elicit their requirements, and express them in PSL. Extrapolating from the inventory project to the projected size of the software market at the time, the PSL/PSA research team calculated that there would not be enough consultants in the world to generalize the approach to the whole industry, or enough money to pay the consultants were they available.

While laymen did not adopt PSL/PSA, it gained proof-of-use among a different population: software professionals. For example, it was adopted by all branches of the Department of Defense, National Dairy (now part of Kraft Foods), Digital Equipment Corporation, IBM, NCR, Burroughs, and Xerox, among others. The challenge of gathering preliminary requirements from large groups of stakeholders nonetheless remained. PSL/PSA was recognized by the French government with the Warnier Prize in 1986 as the first instance of computer-aided software engineering (CASE) tools.

Last Research Mile for Group Support Systems

Proof-of-value and proof-of-use roadblocks on one class of solutions, however, sometimes inspire new solutions for new classes of problems. Teichroew and his team of Ph.D. students had invented CASE, a set of technologies and processes for developing high quality code. The team continued the to make substantial progress on CASE tools for many years [67]. Nunamaker, however, remained focused on the question of how a systems analyst could bring 4000 stakeholders to consensus around the requirements for a large, complex system without having to hire consultants to sit with each stakeholder and write PSL. Nunamaker reasoned that it might be possible to develop software that would let all participants contribute their requirements in plain language in real time to a shared system, and that it might be possible to develop collaboration support to help stakeholders converge, evaluate, organize, and negotiate a final requirement set among the group. The results of that research stream came to be called group support systems (GSS), also known as Electronic Meeting Systems (EMS) and Group Decision Support Systems (GDSS). Early in the research stream, however, it became clear that GSS would not only be useful for software requirements negotiations, but also generally useful for any tasks where multiple people had to work together to create mutually acceptable joint deliverables.

The first proof-of-concept prototype in the GSS research stream was an electronic brainstorming tool built on a VAX mini-computer using the VAX Job Control Language [45]. The prototype started each user on a separate electronic page to which they could add a single idea. After each contribution, the system automatically switched the user to a different page, to which others might already have added ideas. The early tools had only rudimentary features, and were not stable or scalable. They were, however, sufficient to demonstrate the functional feasibility of simultaneous computer-aided deliberation and co-creation of requirements, and were sufficient to support the premise that future implementations might yield valuable gains for groups.

The first GSS tools were originally conceived to support geographically distributed teams connected to the main computer by intelligent terminals. Early trials, however, revealed that people at the time struggled to imagine even the possibility of distributed teamwork. Researchers therefore developed new prototypes based on personal computers connected by a local area network (a radical new technology at the time), and built a first generation of electronic meeting rooms where people could see one another as they worked together and engaged one another through the technology (Figure 4) [51]. Researchers added additional tools to improve the ability of groups to generate, organize, and evaluate ideas [49].

[FIGURE 4 ABOUT HERE]

On the strength of early findings, GSS researchers acquired grant funding to develop more-robust and more-complete proof-of-value prototypes that could be used to support real tasks. Proof-of-value studies in the field provided overwhelming evidence that, under certain conditions, users could benefit by using GSS in the field [53]. Some of the earliest field trials of GSS took place at Boeing and IBM. IBM's Owego plant [50] had 10 years of records about team projects to resolve production line issues. Researchers worked with the line workers to create a new GSS-based work system for production line problem resolution. Analysis revealed a 50% reduction in person hours and a 90% cut in project cycle times over 30 projects, and a triple-digit return on their GSS investment. [34]. Boeing Corporation likewise reported more than 70% savings of labor hours, more than 90% reductions in cycle times, and a triple-digit return on investment for its use of GSS to support concurrent engineering on the 777 project [59].

Proof-of-value findings for GSS-based work systems contradicted beliefs about group work that were widely assumed to be axiomatic. IBM's industrial psychologists, for example, declared the GSS problem-solving work practice to be inappropriate because it allowed anonymous input during brainstorming. Reasoning from the desk chair, they argued that the anonymity must be fostering free-riding, and that the critical comments people made during anonymous brainstorming must also be reducing creativity and damaging group cohesion. GSS researchers therefore conducted controlled experiments to test whether these concerns were consistent with actual outcomes. These studies showed, however, that groups using anonymous electronic brainstorming and a critical evaluative tone produced more unique ideas and more high-quality ideas than did groups using identified brainstorming and a positive evaluative tone [21]. These contributions to knowledge would not have emerged had GSS research ended with proof-of-concept research.

Likewise, it was difficult to publish early GSS findings because reviewers argued from the desk chair that prior research had shown that group performance declined as group sized increased beyond five or six members. GSS researchers, however, were reporting that, under certain conditions, performance of GSS users continued to improve as group size increased to 40 people, which was the largest electronic meeting room available to us at the time. The reviewers argued that the reported outcomes must be the result of poor research or falsified data. Only when multiple researchers from multiple universities obtained similar results were the early findings finally accepted for publication.

Proof-of-value research showed that GSS with a certain feature sets that were used under certain conditions could increase the effects of transformational leadership [65], improve group cohesion [3], creativity [54], and decision quality [46]. GSS benefits could increase with group size [30], with task difficulty [29], and with support from an facilitator [3]. Further exploration showed that performance could be further enhanced by, for example, decomposing complex problems [24], by invoking social comparison (e.g., "Groups who produce fewer ideas than this standard are below average."), and by using humor to raise the salience of a social comparison (e.g. "Groups who produce fewer ideas than this are brain dead.") [64].

Efforts to understand observed effects in GSS gave rise to the Cognitive Network Model of Creativity, which in turn gave rise to new GSS ideation techniques . These and many other findings (see [27, 28] for a compendium of early GSS studies) would not have been made had GSS research ended with Proof-of-concept.

Ongoing proof-of-value research revealed layers of previously unsuspected complexity in the needs of collaborating groups, and those insights, in turn, informed the design and use of increasingly flexible and sophisticated GSS tools. Electronic brainstorming tools, for example, acquired configurable features for more than 20 variables pertaining to ideation, for example, several degrees of participant identifiability, several kinds of idea numbering, and configurable settings to support or limit participant's ability to add, edit, modify, move, delete, undelete, tag, and judge ideas. Idea evaluation tools likewise accreted a range of polling methods in response to a variety of group needs, and a variety of ways to display and respond to the results. Few of these were imagined from the desk chair.

Figure 5 summarizes relationships among some important constructs relating to collaboration that were discovered and documented during the proof-of-value stage of GSS research. These scholarly discoveries could not have been predicted from the desk chair, and would not have emerged had GSS research ended with proof-of-concept.

[FIGURE 5 ABOUT HERE]

Despite strongly positive proof-of-value results in many different domains and work environments, and despite several commercial GSS products being offered in the marketplace, such as Ventana's GroupSystems and IBM's Team Focus (A private-label offering of GroupSystems), GSS did not transition quickly into the workplace [1, 48]. IBM [34] and Boeing Corporation [59] both worked closely with the research community on proof-of-value studies for GroupSystems. Both companies published papers detailing yearlong studies of GroupSystems use in the field that produced gains ranging from 50% to 90%. Both carefully documented triple-digit returns on investment in the first year of their GroupSystems projects. Fortune Magazine ran an article about these remarkable results [41]. Yet, the same week the Fortune magazine article appeared on the newsstands, Boeing canceled its GroupSystems project and reassigned its team of facilitators to other duties. IBM replicated its first single-plant study at five more plants, garnering results that were slightly better than the first study (Grohowski et al, 1990). Yet GroupSystems did not transition widely inside IBM, and for IBM's customers, it rarely spread beyond its initial adoption point.

It became commonplace for organizations to use GroupSystems in successful ways for two-to-three years, to report strong positive benefits from its use, and then to abandon it. The World Bank used GroupSystems actively and effectively in a variety of initiatives to improve conditions in third-world countries [e.g. 40], only to discontinue its use after three years. A two-year demonstration project in D.C. public schools, which at the time had a 64% dropout rate, showed that collaborative learning techniques based on GroupSystems could be used to reduce the dropout rate among learners at risk to almost zero. And yet, despite another year of effort, the research team was not able to persuade any teachers to attempt the new approach in their own classrooms.

Proof-of-value research was not sufficient to drive widespread adoption and diffusion of GroupSystems as it was then conceived. Proof-of-use for certain GroupSystems capabilities nonetheless emerged in ways the proof-of-value researchers did not anticipate from their desk chairs. Many of the features and functions by GSS and other collaboration technology researchers found their way into common use in other kinds of applications, for example, in instant messaging, chat, and social media, and shared editors where multiple people could edit the same document and drawing.   Sophisticated decedents of these tools are now available at no cost from, for example, Google and Zoho. Early versions of GroupSystems pioneered online voting and survey tools, both of which are now standard fare for many online collaboration spaces. GSS and CSCW researchers explored the use of screen sharing, which is now a standard capability for example in Skype and Adobe Connect, and remote desktop capabilities, which are now standard in several operating systems.

Although half-a-dozen companies around the world now sell GSS, they are still a niche product. Proof-of-value was not enough to gain widespread use in the field. Part of the problem may have been that proof-of-value GSS were highly configurable general-purpose collaboration platforms. GroupSystems, for example, supported more than 10^{20} possible configurations (more than 10 million). Each combination had the potential to support multiple patterns of behavior in collaborating groups, and there was nothing inside or outside the system to inform a user about the variety of possible group dynamics, and the effects any given configuration could produce. The benefits of GSS were therefore typically only realized in groups led by collaboration experts – e.g. professional facilitators or GSS researchers. Collaboration experts, however, were not available to many groups, who therefore could not realize the potential benefits of GSS. At that time, much of collaboration expertise was tacit; it was not clear even to the experts what they were doing to produce the observed benefits, nor why they were doing what they did. This spawned a new branch of research called Collaboration Engineering (CE).

The Last Research Mile in Collaboration Engineering

Collaboration Engineering is an approach to designing collaborative work practices for high-value tasks and transferring them to practitioners to execute for themselves without ongoing support from collaboration experts [12, 70]. In moving the approach through the last research mile, researchers made a number of conceptual, theoretical, methodological, and technological contributions. One of us, (Briggs) worked with G.J. de Vreede and several graduate students to develop a design pattern language called ‘ThinkLets,’ a set of named, scripted collaboration techniques that caused predictable effects on group dynamics. Proof-of-concept ThinkLets turned out to be useful building blocks for designing collaborative work practices, and useful training tools for transferring a work practices to others [9, 12, 69]. ThinkLets also offered an explanation for conflicting results in the GSS literature: different experimenters used the same technology, but different techniques, and so obtained different results [62]. Proof-of-concept ThinkLets, however, were technology dependent; each was tied to a specific configuration of GroupSystems.   Researchers reimagined ThinkLets as logical design elements with technology-independent specifications [42]. The core element of the new specification represented essence of a technique with abstract rules using this template:

> *A person in some* ROLE *takes some* ACTION *using some* CAPABILITY *under some* CONSTRAINTS.

The ACTIONS in each rule could be expressed with five primitives: Add, Edit, Move, Delete, and Judge [42]. For example, Osborne’s classic brainstorming technique could be expressed:

> A PARTICIPANT (role) ADDs (action) ideas to an SHARED AUDIO CHANNEL (capability) under the following constraints: a) focus on quantity; b) withhold criticism; c) welcome unusual ideas; and d) combine and improve ideas.

So long as the CE chose a technology that provided the capabilities required by the rules, and wrote a script that instantiated the rules, a ThinkLet would produce predictable, repeatable group dynamics, regardless of whether it was implemented with software or whiteboards or paper. ThinkLets also made it possible to document group activities concisely. Before ThinkLets, it took 170 pages to document a collaborative requirements negotiation methodology called EasyWinWin [8, 35, 36]. Two years later, it took only two pages to document EasyWinWin in ThinkLets notation. Facilitators who knew the ThinkLets, but were unfamiliar with EasyWinWin, were nonetheless able to execute the process without further training.

A second breakthrough in CE came when researchers identified and defined six patterns-of-collaboration that characterized the ways groups moved through their activities [70]. Each pattern defines a change-of-state produced by group effort:

1. Generate: move from fewer to more concepts in the shard set
2. Reduce: move from many concepts to fewer deemed worthy of more attention
3. Clarify: move from less to more shared understanding of concepts
4. Organize: move from less to more understanding of relationships among concepts
5. Evaluate: move from less to more understanding of the utility of concepts for goal attainment
6. Build Commitment: move from fewer to more people willing to commit to a proposal

ThinkLets, it turned out, were, techniques for invoking predictable variations of these six patterns. Some brainstorming techniques, for example, push a group for depth and detail on a narrow set of topics, while others foster breadth and variety, while blocking participants from thinking deeply. Researchers developed ways to measure and study each of the six patterns [e.g. 5, 20, 22, 43, 44, 61, 68], and began to develop causal theories to explain observed variations in these patterns, for example, theories of ideation quality [17], creativity [63], and building consensus [10]. The six patterns became logical design components for mapping the path a group should take through its activities before choosing which techniques to use. To better understand how ThinkLets-based collaborative work systems could be transitioned to practitioners, they also developed theories to explain, for example, willingness-to-change [15] and satisfaction responses [18].

In 2008, CE researchers created a seven-layer model of collaboration (SLMC) [11, 60] where the six patterns of collaboration and ThinkLets appeared in Layers 4 and 5 and collaboration technology appeared as Layer 6. Each layer considered collaboration at a different level of abstraction:

1. Collaboration Goals – addresses group goals, private goals, and goal congruence
2. Group Deliverables – Work products by which to achieve goals
3. Group Activities – work breakdown structure for creating deliverables
4. Patterns of collaboration – changes-of-state characterizing how groups move through activities
5. Collaboration Techniques – ways to invoke variations on the patterns of collaboration
6. Tools – technologies for instantiating the capabilities required by each technique
7. Behaviors – things people say and do with tools to instantiate techniques

For the CE research community, SLMC provided an organizing structure for many of the phenomena, concepts, theories, best practices, techniques, and metrics pertaining to each layer. Thus, it offered a backbone for organizing the knowledge of Collaboration Engineering as a scholarly discipline. For collaboration engineers it provided a separation of concerns to reduce cognitive load and improve completeness-of-work-products during collaboration engineering projects [14]. Recently, SLMC also became the backbone for an second-generation CE design methodology of several stages, with goals, deliverables, key activities, and quality indicators for evaluating the performance of collaboration engineers on each key activity [16].

While ThinkLets made it faster to train non-experts to create value with collaboration technology, the problem remained that untrained users still could not realize the potential benefits of the technology. CE researchers therefore took on a new research challenge: *to package collaboration expertise with collaboration technology in a form that non-experts could use successfully with no training on either techniques or the technology*. We reasoned that it might be possible for non-experts to execute a well-engineered technology-supported collaborative work practice with no training if the software were designed with capabilities and embedded guidance that exactly matched a specific task. This spawned yet another branch on the research tree that began in the 1960's with PSL/PSA: Collaboration Support Systems (CSS).

In a CSS, a collaboration engineer snaps together Process Support Applications (PSA), using small, loosely coupled, highly configurable collaborative components to create tools that exactly match the structure of a group's task without having to write new software code [13]. A PSA moves a group through a sequence of activities designed by an expert. Each activity presents users with just the tools they need, configured to support just the right actions and to display just the right data to just the right people and with just the right guidance they need to complete the activity successfully, and presents them with nothing else. Proof-of-concept prototypes showed that the CSS approach could cut development times for real-time collaboration systems by two orders of magnitude. Proof-of-value prototypes demonstrated that non-expert practitioners could able to successfully execute an engineered work practice with no training on either the collaboration technologies or collaboration techniques [13]. Proof-of-value research continues on CSS.

The new constructs, relationships, theories, models, techniques, methods and technologies that emerged from proof-of-value and proof-of-use research on GSS, CE, and CSS could not have been contributed to the scientific literature had PSL/PSA research or later GSS research ended with proof-of-concept or proof-of-value. Most practitioners would not have been able to derive value from those inventions.

The Last Research Mile for Credibility Assessment Technology

As practitioners started using collaboration technology for tasks with higher risks and higher stakes (e.g. homeland security, counter-terrorism, crime prevention), they began to question how they could evaluate the credibility of the contributions people made to the systems. How credible was the information? Was someone trying to divert the agenda? Proof-of-value and proof-of-use research on GSS therefore spawned yet another branch of productive last-mile research: multimodal rapid credibility screening technology.

Deception is defined as a message knowingly transmitted by a sender to foster a false belief or conclusion by the receiver (Burgoon, 1996). Research shows that humans produce dozens of linguistic, behavioral, and physiological cues when they intend to deceive. Linguistically, for example, deceivers tend to use fewer first-person and third-person pronouns (referencing individuals), but more second-person pronouns (reference groups), which may signal a desire to distance the deceiver from the deceptive language [37]. Behaviorally, people tend to move their hands, heads, and bodies differently when deceiving than when telling the truth. They tend, for example, to use fewer illustrating gestures, and tend to hold their torsos more rigid when deceiving [55]. Physiologically, attempts to deceive trigger, for example, dilation of pupils, accompanied by increases of blood pressure, heart and respiration rates [31].

Deceivers can consciously control three-to-five deception cues in real time, so no single cue can serve as a reliable indicator of deception. The uncontrolled cues, however, leak into the environment and may be detected. Two of us (Nunamaker and Derrick), working with Judee Burgoon and graduate students, began exploring how many cues-to-deception could be automatically and reliably detected in written and interpersonal communication. The goal was to find a manageable set of about 15 reliable deception cues that could be detected non-invasively (without attaching sensors to subjects) and analyzed in real time to offer real-time support in the field for high-stakes tasks such as counter-terrorism, drug interdiction, and border security. Those tasks required that agents assess the credibility of very large numbers of people, for example, as they disembarked from airplanes or as they crossed a border checkpoint. It is not economically or operationally feasible to run polygraph tests on each subject in those conditions.

The first proof-of-concept prototype was a lab equipped with a chair for a subject, surrounded by an array of non-invasive sensors, for example, microphones, thermal imaging, ultrafast infrared and near infrared cameras, high-definition visible-spectrum video cameras, and a laser Doppler vibrometer (LDV) that could detect a subjects heart rate and respiration rates at a distance (Figure 6).

[FIGURE 6 ABOUT HERE]

Proof-of-concept research focused on assessing the technical feasibility of real-time credibility assessment for a) kinesic cues, such as gestures, micro-movements, eye-gaze, and body rigidity; vocalic cues such as speech duration, time-to-response when a question was asked, and latency; voice pitch and amplitude; and physiological cues: blood pressure, heart and respiration rate, facial skin temperature, and facial pore size (Derrick, et al. 2010). Early research showed that certain vocalic cues reliably correlated strongly with intent to deceive. For instance, when people were attempting deception, not only did the pitch (frequency) of the voice rise and become more varied, but also the periods during which the loudness of the voice did not change grew longer and more varied in length. The research also showed that the LDV was effective at detecting pulse and respiration rate, and that slower rates of inter-beat deceleration were associated with attempts to deceive (Derrick, et al. 2010). However, normal movements of the subjects made it difficult to keep the LDV accurately trained on the part of the neck where data could be collected. If the subject had a beard or was wearing a turtleneck sweater, the LDV had a difficult time generating a good signal.

Proof-of-concept studies also showed that thermal imaging could be used to detect increases of skin-temperature, and that, as other research had shown, deception was strongly associated with increased skin temperature, especially for the skin around the eyes. Ambient conditions outside the lab, however, – e.g. sunlight through windows, heating and air conditioning vent – tended to confound the heat sensors. The changes in temperature, however, were caused by increased blood flow to the skin, and researchers found that high-resolution video cameras were sufficiently sensitive to pick up the changes in skin color associated with increased blood flow. Those cameras also made it possible to map the dilation of skin pores in real time. Skin pore dilation turned out to be a stronger predictor of deception than was change of skin color.

The team added eye-tracking sensors to the sensor suite, and determined they could readily detect increased blink rates, startle responses, pupil dilation, each of which was associated with deception in their studies. They also found that guilty people tended to focus their gaze for significantly longer periods on images related to their guilt than when they viewed irrelevant images. For example, in one study, some subjects were asked to assemble a simulated pipe bomb. Later, all subjects were shown an image of the pipe bomb with one key component missing. Those who had assembled a pipe bomb focused their gazes on the location of the missing element a greater number of times and gazes on that location were of longer duration (Derrick, Jenkins, & Nunamaker, 2011).

Before the team could test the solution in the field, they had to find a way to package the sensor suite along with the computers required to analyze the signals into a unit sufficiently small and robust that it could be taken into the field to support real work. They designed the second proof-of-concept prototype into a rack of laboratory shelves about three feet wide and six feet high, and configured it to be used by a person who stood in front of it. The new prototype included a screen at eye level to display an animated avatar that would ask the users a set of pre-determined questions, like:

- Please describe in detail the contents of your backpack or purse.
- What will you do after you get through this checkpoint?
- Has anybody given you a prohibited substance to transport through this checkpoint?
- Do any of these items in the bag not belong to you? If so, please describe which items those are.

If the sensors revealed cues known to be associated with deception, the avatar would respond with pre-determined answers like:

- I am detecting deception in your responses. Please explain why that is.
- Why should I believe you?
- What should happen to a person that unlawfully takes prohibited substances through a checkpoint?
- How do you think that our assessment of your credibility will work out for you today?

The prototype included a high-definition video camera, a FLIR thermal camera, an ultrafast near Infrared camera, a pupillometric near-infrared camera, an eye tracker, and an LDV. The team reasoned that a user standing close to the system and responding to a screen might be less likely to move in ways that would disrupt the LDV.

To support the research, the avatar could be configured to present as male or female, and could be configured in varying sizes with varying voices and accents, ages, emotional demeanors, ethnicities, hairstyles and hair colors, clothing, and facial structures (Figure 7). In laboratory studies, subjects perceived male avatars to be more powerful, more expert, and more trustworthy than female avatars. Female avatars, in turn, were perceived to be more likeable [47]. Responses to differences in the apparent age of the avatar varied by culture.

[FIGURE 7 ABOUT HERE]

It was not possible from the desk chair to guess how people might respond to being questioned by the avatar. It might have been that people would react with incredulity and give frivolous responses, or that that they might dismiss the avatar completely. The team was surprised to discover that, instead, people treated the avatar with respectful behaviors similar to those typically accorded to an authority figure. Fifteen percent responded, for example, with phrases like “Yes, sir” and “No, ma’am. In a curious turn of events, several participants in a mock-crime study confessed to real felonies under the questioning of the technology while overseen by a practicing polygrapher. Researchers had to suspend the project for two weeks to work out how to respond to such confessions. It would be unethical to continue the research if the experiments were going to prompt subjects to do things harmful to themselves

To gain proof-of-value, researchers created a proof-of-value prototype packaged in a movable kiosk similar in size to an automated teller machine (Figure 8). In the field, subjects tended to interrupt the avatar to answer a question before it finished asking. The combined signal of the avatar's voice and the subject's voice made it difficult to interpret the vocalic signals of the subject, an outcome they did not anticipate from the desk chair. The team therefore added a touch-screen to the system at waist level with virtual button users could press to talk when the avatar finished speaking. The screen could also present images for guilty-knowledge tests. The button distracted some users, however, who tried to press it at the wrong time, and then puzzled over why the button had no effect. Later versions of the system eliminated the second screen.

Some of the earliest proof-of-value trials of kiosk took place at border crossing on the USA-Mexico border. The kiosk was configured to speak Spanish to users. However, some subjects interpreted the avatar to be hostile, arrogant, and difficult to understand because it used 12^{th} grade-level language and spoke with a Castilian, rather than Mexican accent. Researchers reconfigured the avatar with 6^{th}-grade level language and the local accent, and the system was better received.

Researchers also discovered in the field that unanticipated ambient noise – e.g. train whistles, crying children, background conversations – interfered with analysis of vocalic cues. They learned that if they switched to high quality directional microphones and used a second microphone to support digital filtering of ambient noise, the problem could be eliminated.   The team found that they could not detect deception by comparing an individual's linguistic, vocalic, kinesic, and physiological readings against an absolute baseline. There were wide variations as a given person moved from context to context, and marked differences from person to person and from culture to culture. Some cultures, for example, tend to use many more and broader illustrating gestures than others. The team had to configure the system to capture baseline readings for each metric for each individual at the start of the interview, then compare the metrics against that baseline with each important question. It was changes-of-state, rather than absolute states that predicted intention to deceive. The research also found that widely used heuristics for detecting deception did not hold. For example, despite a widely held belief among practitioners in the field that lack-of-eye-contact predicted deception, field research found that eye contact was no better than random at indicating deception.

In laboratory studies where ground truth was known, expert polygraphists were able to identify test subjects who committed a mock-crime with an average accuracy of about 60-70% accuracy. The new avatar system attained an average accuracy of 80-90% [52]. Outcomes like these provided compelling proof of value, but there was more to be learned in proof-of-use research.

At Nogales, Arizona a field test was conducted for the Trusted Traveler program. The last step in the process of obtaining a Trusted Traveler card requires that each applicant submit to a personal interview. The interview consists of a fixed set of questions that are asked by an officer. However, the officers uniformly reported that the interviews are the most boring part of their work. They also reported being understaffed, so this seemed a good task for the AVATAR/Kiosk. The officers typically, did not ask the full set of questions and they routinely clarified the wording and intent of the questions. Given their staffing shortages and heartfelt distaste for the interview protocol, researchers assumed from the desk chair that the experts would quickly accept and deploy the kiosk in the field. The kiosk could ask the 20 questions accurately and in the same order every time. The AVATAR would send a report to the officer with an indication of which applicants might be attempting deception, and would refer potential deceivers to the officers to do what officers said they most enjoyed: informal, creative and persistent questioning to tease out the truth.

When the proof-of-use prototype went live in the field, executives praised it, and gave every indication that they would accept it. The officers also stated that they liked the Kiosk, but some officers declined to use it, even though they disliked doing the interviews. Emotionally, the officers self-identified as the people who couldn't be fooled. They stated emphatically that there was no possibility a machine could do better than they. Politically and economically, some officers were concerned that the system might do more than just screen applicants – it might take their jobs.

Proof-of-use research continues on the credibility assessment kiosks. Studies are currently underway on three continents to discover and document the degree to which people can derive value from the solution across a variety of high-stakes context. These studies continue to reveal new technical, economic, and operational feasibility issues that could not have been imagined when the solution was first conceived from the desk chair. When this research stream ends, not only will there be a new, useful, generalizable solutions for credibility assessment challenges that span many domains, but there will also be a richer, more detailed body of scholarly work reporting exploratory, theoretical, and experimental foundation for phenomena related to deception and deception detection.

Discussion and Conclusions

The insights that come from the last research mile cannot be obtained from the desk chair. Impact of the research is maximized as one traverse the last research mile. When it comes to technology transition, the devil is in the details, and the details are in the field. An interesting idea for a new solution will contribute very little to scholarly knowledge and will make little impact on society until someone takes it through the last research mile. A research community's understanding of a problem domain will remain rudimentary until they have moved solutions through the last research mile. Proof-of-concept technologies are impoverished compared to the proof-of-use systems.

The Scientific Utility of the Last Research Mile

From a distance, some might worry that the activities of last-mile research seem uncomfortably similar to rapid prototyping as conducted in conventional systems analysis projects. They might question, "Is this really research?" There are, indeed, differences at every level between ordinary systems analysis and last-mile research. These differences are perhaps most easily illustrated by contrasting the deliverables of the systems analysis with the scholarly products of last-mile research. A systems analyst seeks to provide a specific solution to a specific problem for a particular group of people in a specific context. Such solutions need not be novel, generalizable or rigorously validated. By contrast, a last-mile researcher seeks to discover, describe and empirically validate the existence of *an important class of important unsolved problems* in the field [19]. A last-mile researcher gathers data to synthesize and publish *exploratory models of observed correlations* among constructs relevant to the problem domain, and the conditions under which those correlations manifest in various contexts. [66]. A last-mile researcher brings the existing scientific literature to bear to predict and explain the effects design choices in light of such factors [39]. When the literature does not fully explain observed phenomena, a last-mile researcher develops and tests *new theories* to better understand constructs that may support or interfere with successful deployment of proposed solutions [e.g. The Yield Shift Theory of Satisfaction, 18, 32]. Where a systems analyst negotiates requirements for a specific solution, the last mile research derives and publishes *generalizable requirements* for solutions to the class of problems under study, and creates *new*, *generalizable solutions* that can be applied across many contexts [19].

Last mile researchers develop *expository instances* of general solutions and use rigorous qualitative and quantitative methods to *validate* that a prototype system achieves its design goals [33]. A last-mile researcher seeks further to contribute back to the literature a *design theory* for the generalizable solution – a concise, validated body of knowledge that practitioners can use to create their own successful instances of the generalizable solution [33].

On the Difference between Ideas and Last-Mile Achievements

Occasionally, systems researchers express reluctance to share their knowledge with colleagues for fear others may "steal their ideas." Ideas are a dime-a-dozen. Last-mile researchers accrete rich, integrated bodies of tacit and explicit knowledge pertaining to the scientific technical, economic, and operational feasibility of intractable problems and technical solutions. From that foundation, such researchers regularly map career-scale research agendas with their doctoral students and colleagues. There is no shortage of ideas in the last research mile; there is only a shortage of people who can act to achieve proof-of-use, to the benefit of society. Others may emulate a particular solution, or even pick up and run with an idea they hear, but unless they also travel the last research mile, they will not be able to emulate the continuous flow of scholarly and technical achievements a last-mile researcher produces, nor conceive of the non-intuitive improvements that follow from the solution they emulate. The advantages derived from one's last research mile achievements cannot be stolen.

Rigor vs. Relevance: A False Dilemma

The last research mile negates the debate in systems fields that technical researchers must trade off rigor and relevance, showing it to be a false dilemma. *Rigor* is the degree to which research practices follow the standards of logic dictated by the epistemology under which it claims to have contributed new knowledge. *Relevance* is the degree to which research contributes directly to improving outcomes of interest to practitioners in the field, i.e. solves an important class of problems. In the last research mile, exploratory research to discover and describe important classes of unsolved problems in the field can be conducted with full exploratory rigor while remaining fully relevant to the workplace. A last-mile researcher might use all the standards of rigor of theoretical research to derive a theory to predict and explain observed variations in outcomes of interest to practitioners, and use that theory to inform design choices for a high-impact systems solution, rendering the rigorous theory fully relevant to the workplace. An experiment that contrasts a solution informed by theoretical logic with an earlier solution can be simultaneously a rigorous test of the theory and a relevant validation of the new solution. A researcher who travels the last mile can therefore leave behind a legacy of fully rigorous scholarly contributions that are simultaneously directly relevant to solution stakeholders.

Consider, for example, this small case. A facilitator worked with executives in a multinational financial institution facing a rapidly changing market and increasing competition. The executives generated a set of 675 paragraph-length strategic issues they deemed critical to the ongoing survival of the organization. They developed and carefully defined 75 categories to classify the issues. Their collaboration system let them drag-and-drop the issues from a list in one column into categories in the next column. Cognitive load was so high, however, that after two hours, the executives, angry and exhausted, abandoned the effort. The categories were carefully labeled with two-sentence-long rigorous descriptors. It was too difficult to find the category they needed among the list seventy-five.

Researchers turned to the Time Based Resource Sharing Model of Working Memory (TBRS) [6], a new and somewhat controversial theory that proposed a causal explanation for cognitive load. Drawing on TBRS, one of us (Briggs) and his graduate students devised a simple new idea-organizing technique, which, if the theory held, should minimize cognitive load. The theory proposed that human working memory is single threaded – capable of executing only one operation at a time. The theory proposed that a small set of primitive operations that comprise the functions of working memory. Three among them are push-to-long-term-memory, fetch-from-long-term-memory, and refresh-working-memory. The team explored how many words a person could typically read at one glance, corresponding to one fetch-from-long-term-memory. It turned out to be about three words. They therefore reduced all category names to no more than three words. The long, rigorous descriptions of each category were made available at a mouse click, should a user need to verify the formal definition of the short label. Next, working memory typically can only manipulate a small set of concepts simultaneously. Researches therefore organized the seventy-five categories under 7 super categories. Where more than seven original categories appeared under a super category, those were further organized under a second layer of super-categories, so that there were no more than 7 original categories on any branch of the category tree. Researchers then instituted a cascading sort procedure. The participants first sorted the 675 issues into the seven super-categories. Then, one category at a time, the group sorted the issues into the second layer of super-categories. Finally, taking one second-layer category at a time, they organized the issues into the original categories. Thus, at no time did a participant need to choose among more than 7 categories to place an issue.

A pilot experiment contrasting the prior organizing technique with the new theory-driven solution showed that participants could organize the 675 ideas into the 75 categories in less than 10 minutes with no reports of cognitive load and no negative emotions. If such a study were conducted with due experimental rigor, the study would not only validate the efficacy of the cascading sort solution, making it fully relevant to the stakeholders, but would also be a rigorous experimental test of TBRS, the theory that informed design choices for the new solution. Thus, by traveling the last research mile, IS researchers can produce works that are simultaneously rigorous and relevant.

Closing Thoughts on the Last Research Mile

While technology-focused research begins from the desk chair, it cannot progress far until it moves through the last research mile. Proof-of-concept research demonstrates the existence of an important class of unsolved problems in the field, and demonstrates the technical feasibility of a generalizable solution. Proof-of-value research uses lab and field studies to demonstrate empirically the potential efficacy of a generalizable solution. The research is not complete, however, until proof-of-use research demonstrates that a self-sustaining and growing communities of practice has emerged around the solution. Until then, a vast body of scholarly knowledge lies undiscovered and uncreated.

Each phase of the last research mile offers opportunities to make scholarly contributions that are fully rigorous and fully relevant. The scope and scale of those opportunities, however, increases geometrically with each stage. Proof-of-concept tends to produce a few modest, yet interesting insights. Proof of value research tends to accelerate and produce a number of exciting technical, operational and scholarly breakthroughs. Proof-of-use research, though, tends to produce a rich, diverse, and integrated body of scholarly knowledge pertaining to the phenomena relevant to the problem and solution spaces.

Disciplines whose researchers travel the last research mile also benefit, gaining stature in the academic community and garnering the respect of their reference disciplines. It is the way to maximize societal impact for systems research. In turn, researchers who travel the last research mile can gain access to more research resources, create more scholarly knowledge, and earn recognition in their fields as pioneers and leaders. More importantly, perhaps, they may see empirical evidence that the world is just a little better off because they were there.

There is no down side, and the first step is easy. Choose a real class of unsolved problems in the field. One need not know how to solve the problem. Were a solution handy, research would not be required. Jump in up to your elbows and see what happens.

Citations

Table 1. Dimensions of Value **(Adapted from Briggs et al. 1999)** **Technology-based systems exist to create value for their stakeholders. Individuals perceive value along a number of dimensions, among them those defined in this table. A perceived value has both a magnitude and a direction (positive or negative). Systems solutions therefore require more than technical and economic feasibility. They must also be operationally feasible, which is to say, politically, social, cognitively, emotionally and physically acceptable to their stakeholders.**	
Economic value	Related to the production, distribution and use of monetary and material resources
Political value	Related to the distribution power and influence among stakeholders
Social value	Pertaining to relationships among individuals, groups, and society. Encompasses issues ranging from a need for companionship, to social rank and status, to the maintenance of civic order.
Cognitive value	Associated with the limits of and demands upon human attention resources and reasoning ability.
Emotional value	Related to the onset, magnitude, duration and valence of one's affective responses, and the interpretations one attributes to those responses.
Physical value	Pertaining to health, fitness, safety, comfort, and well-being of a person's body.

Figure 1. Collaboration between financial advisor and client using the tabletop advisory system. Shared widgets on the tabletop improved process and information transparency. Users attend to two distinct spaces during the advisory process – a relationship-building space where they make eye-contact and create shared understanding; and the artifact work space where they view and manipulate screen objects. Clients who used the system understood the process better and perceived greater control over it and over the process. They perceived advisors to be more trustworthy. These correlated with increased satisfaction. Adapted from (Heinrich et al. 2014).

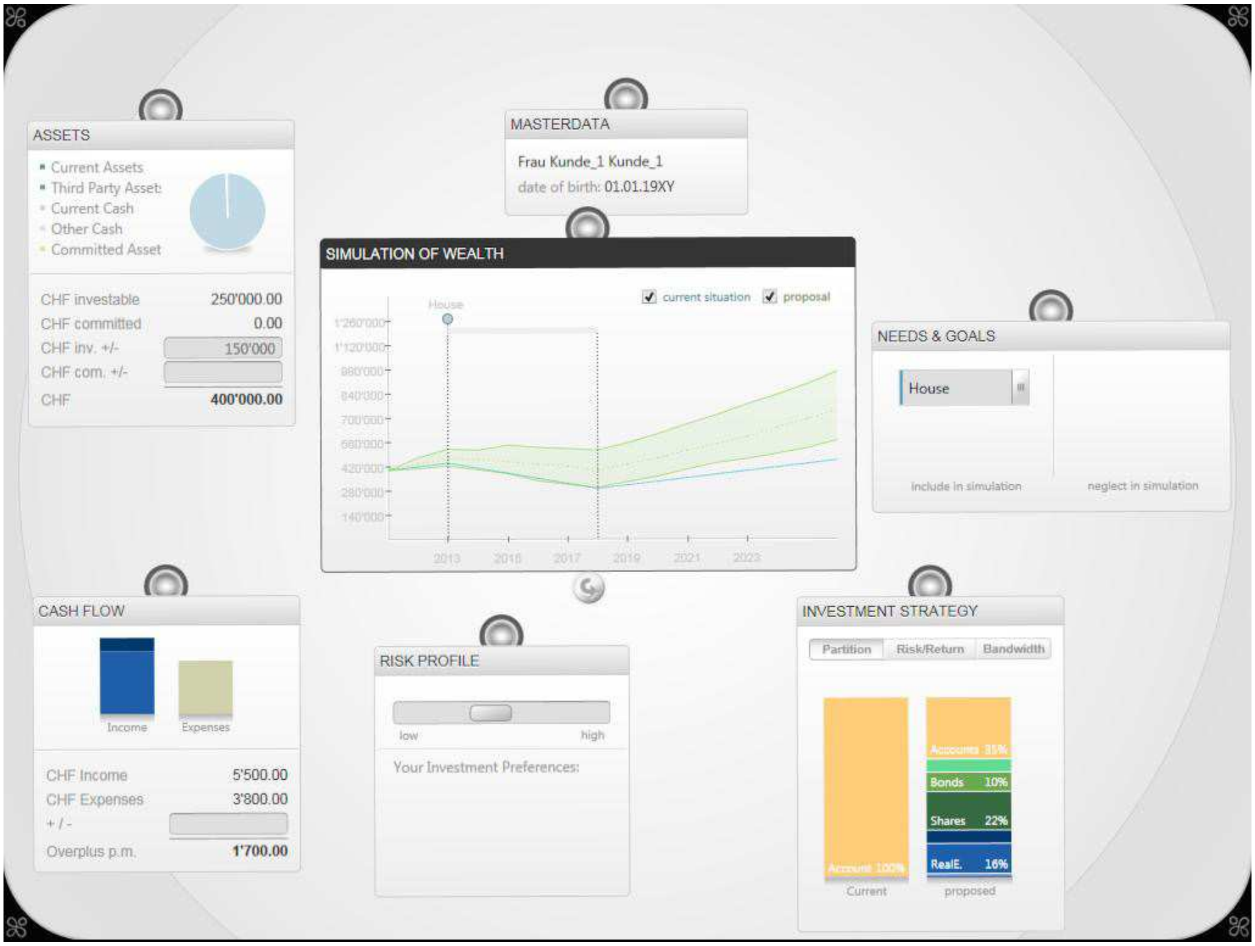

Figure 2. A table-top financial advisory system with widgets for each aspect of the advisory process. All widgets operate on the same data set, so changes in one widget reflect in other relevant widgets. Users can drill down to see the details underlying the display in any widget. Widgets may be rotated with the fingers and passed from user to user as with papers on a physical table, so the advisor and client co-control the process flow [57].

Table 2. Summary of the Last Research Mile in Three Phases			
	Proof of concept	**Proof of Value**	**Proof of Use**
Goals	• Establish technical feasibility • Understand the problem space • Discover/describe Phenomena of Interest and their correlates in the problem space	• Accurately measure the efficacy of solutions • Discover/describe unintended consequences • Design processes using system to create value • Better understand of feasibility issues	• Create self-sustaining and growing communities of practice around a solution • Codify knowledge practitioners need to build their own instances of a generalizable solution
Prototypes	• Functionality sufficient to test technical feasibility for simple tasks, but not for real work • Rudimentary, not necessarily scalable, full featured, stable	• Sufficient functionality to solve at least one real problem • Sufficiently robust to survive field trials with real users doing real work • May still need improvisation, work-arounds	• Sufficiently full-feature to support to support every-day real work process • Sufficiently robust to run unattended in the work place without support from researchers
Typical Research products	• Detailed problem descriptions • Generalizable requirements • Models of observed correlations • Theories to explain and predict observed correlations • Course-grained metrics for Phenomena of interest	• New phenomena-of-interest and their correlates. • New theoretical logic to explain observed phenomena • Generalizable requirements • Generalizable solutions • Exemplar instances of solutions • Rigorous metrics for solution efficacy • Empirical evidence of solution efficacy	• Definitive problem statement • Definitions of key constructs • Theories that inform design choices • Generalizable requirements for solutions • Principles of form and function • Design methodologies for solutions • Exemplar instances of the solution • Rigorous experimental & field tests of theories, solutions • Metrics of successful implementation
Typical Research Methods & Standards of Rigor	• Exploratory case studies with concatenation across contexts, conditions • Engineering research	• Exploratory research with concatenation across contexts, conditions • Experimental research with rigorous design and metrics • Theoretical Research • Engineering research	• Exploratory research with concatenation across contexts, conditions • Experimental research with rigorous design and metrics • Theoretical research • Engineering research
Advantages to the Researcher	• Landmark publications about: ○ The problem space ○ Generalizable requirements ○ Potential solutions ○ Discovered phenomena and their correlates • Increased credibility with grant sponsors	• Rich body of explicit and tacit knowledge about the problem and solution spaces • Grant proposals with a concrete specificity not available to researchers who have not engaged in proof-of-value research • Rich variety of exploratory, theoretical, experimental, and applied science publications	• Deep, sophisticated understanding of technical, economic, and operational aspects of the problem and solution spaces. • Find new nuggets of knowledge on purpose • Create value intentionally for problem owners • Participate in commercialization revenues • New resources to advance further research

```
       PSA Version A4.0R2                                          JUN 29, 1976   11:28:44       Page
                                       University of Michigan - MTS

                                        Formatted Problem Statement

Parameters:   DB=-EXADB[1] FILE=-PSANAME  NOINDEX  NOPUNCHED-NAMES  PRINT' EMPTY  NOPUNCH  SMARG=5
  NMARG=20  AMARG=10  BMARG=25  RNMARG=70  CMARG=1  HMARG=60  NODESIGNATE  ONE-PER-LINE  DEFINE
  COMMENT  NONEW-PAGE  NONEW-LINE  NOALL-STATEMENTS  COMPLEMENTARY-STATEMENTS  LINE-NUMBERS
  PRINTEOF  DLC-COMMENT

INPUT                                                                 employee-information;
       /*    DATE OF LAST CHANGE - JUN 26, 1976, 15:27:33 */[2]
     GENERATED BY:  departments-and-employees;
     RECEIVED BY:   payroll-processing;

INTERFACE                                                             departments-and-employees;
       /*    DATE OF LAST CHANGE - JUN 26, 1976, 11:11:57 */
     GENERATES:     employee-information;
     RECEIVES:      paysystem-outputs;

OUTPUT                                                                paysystem-outputs;
       /*    DATE OF LAST CHANGE - JUN 26, 1976, 15:27:33 */
     GENERATED BY:  payroll-processing;
     RECEIVED BY:   departments-and-employees;

PROCESS                                                               payroll-processing;
       /*    DATE OF LAST CHANGE - JUN 26, 1976, 11:11:57 */
     GENERATES:     paysystem-outputs;
     RECEIVES:      employee-information;
     UPDATES:       payroll-master-information;

SET                                                                   payroll-master-information;
       /*    DATE OF LAST CHANGE - JUN 26, 1976, 11:11:57 */
     UPDATED BY:    payroll-processing;

EOF EOF EOF EOF
```

Figure 3. Problem Statement Language (PSL), a structured English format for software requirements. PSL requirements could be fed into the Problem Statement Analyzer (PSA) which could check them for completeness, consistency, and correctness. In 1972, 4000 stakeholders on a large-scale software development project refused to express their requirements in PSL. PSL/PSA, therefore, failed Proof-of-value trails. That failure spawned 40 years of high-visibility research that might have been lost had research on PSL/PSA stopped with Proof-of-concept trials.

Figure 4. The first-generation electronic meeting room at University of Arizona. In the early 1980's stakeholders in large-scale software development projects struggled to imagine the possibility of same-time-different-place collaboration, and so declined to use early proof-of-value prototypes of group support systems. Researchers therefore developed a same-time-same-place electronic meeting room to support proof-of-value research. The lab was called the "DIC." The original meaning of the acronym is lost in the mists of time. A decade later, same-time-different-place and different-time-different place collaboration were common.

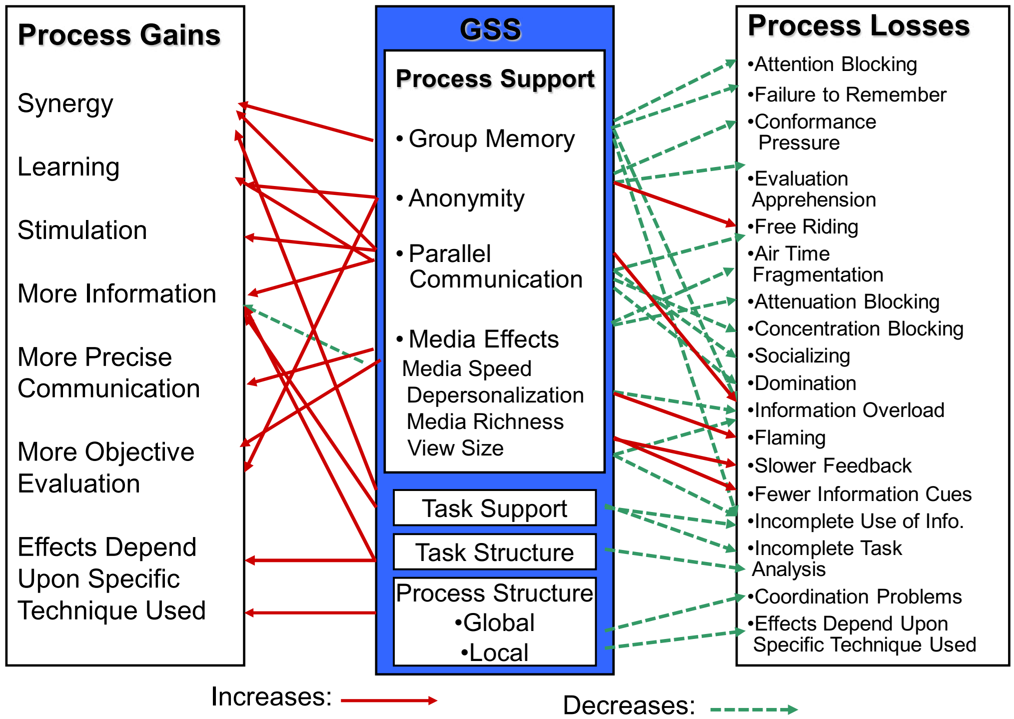

Figure 5. A Model of GSS Functionality in Terms of the Process Gains and Losses. The model is derived from Proof-of-value research that would not have been possible with a Proof-of-concept prototype. Adapted from (Nunamaker, et al, 1991).

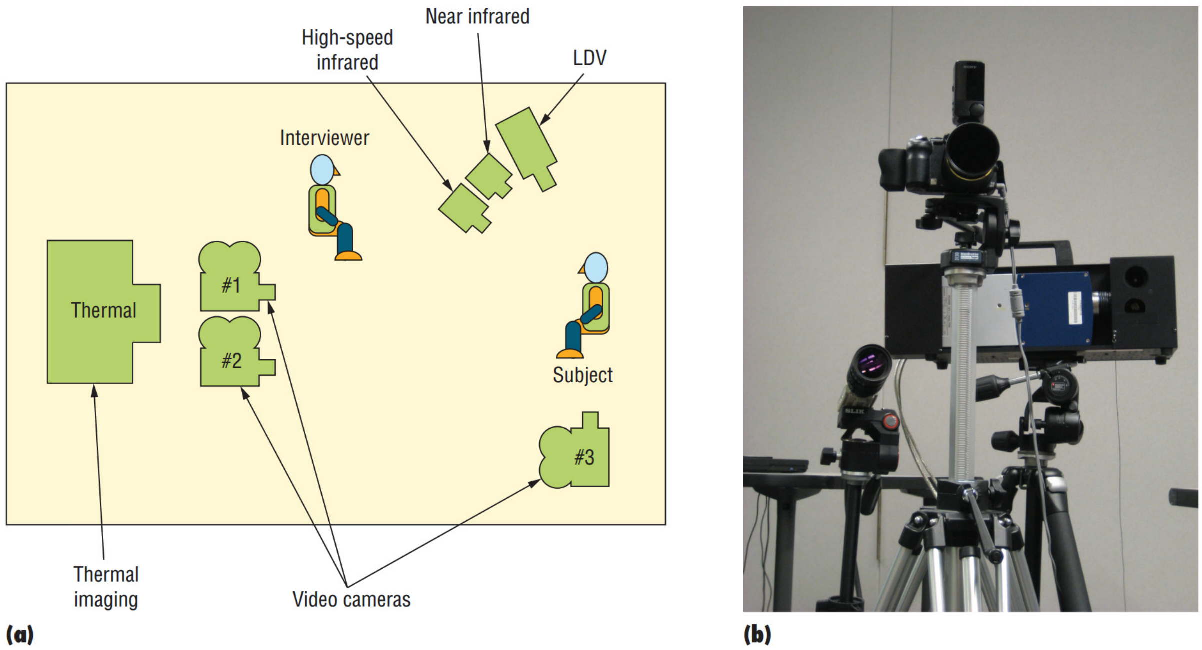

Figure 6. First Proof-of-Concept Prototype for Real-time Non-invasive Detection and Analysis of Multiple Cues to Deception. The Credibility Assessment Lab provided a chair for an experimental subject surrounded by an array of sensors for detecting deception cues (a). These included a thermal imaging system, an ultrahigh-speed infrared camera, a near infrared camera, several high definition visible-spectrum cameras, and a laser Doppler vibrometer (LDV) that could remotely detect heart rate and blood pressure (b). [Adapted from 25]

Figure 7. Configurable Avatars for the Credibility Assessment system. The avatars could present as male or female, and could be configured in varying sizes with varying voices and accents, emotional demeanors, ethnicities, hairstyles and hair colors, clothing, and facial structures.

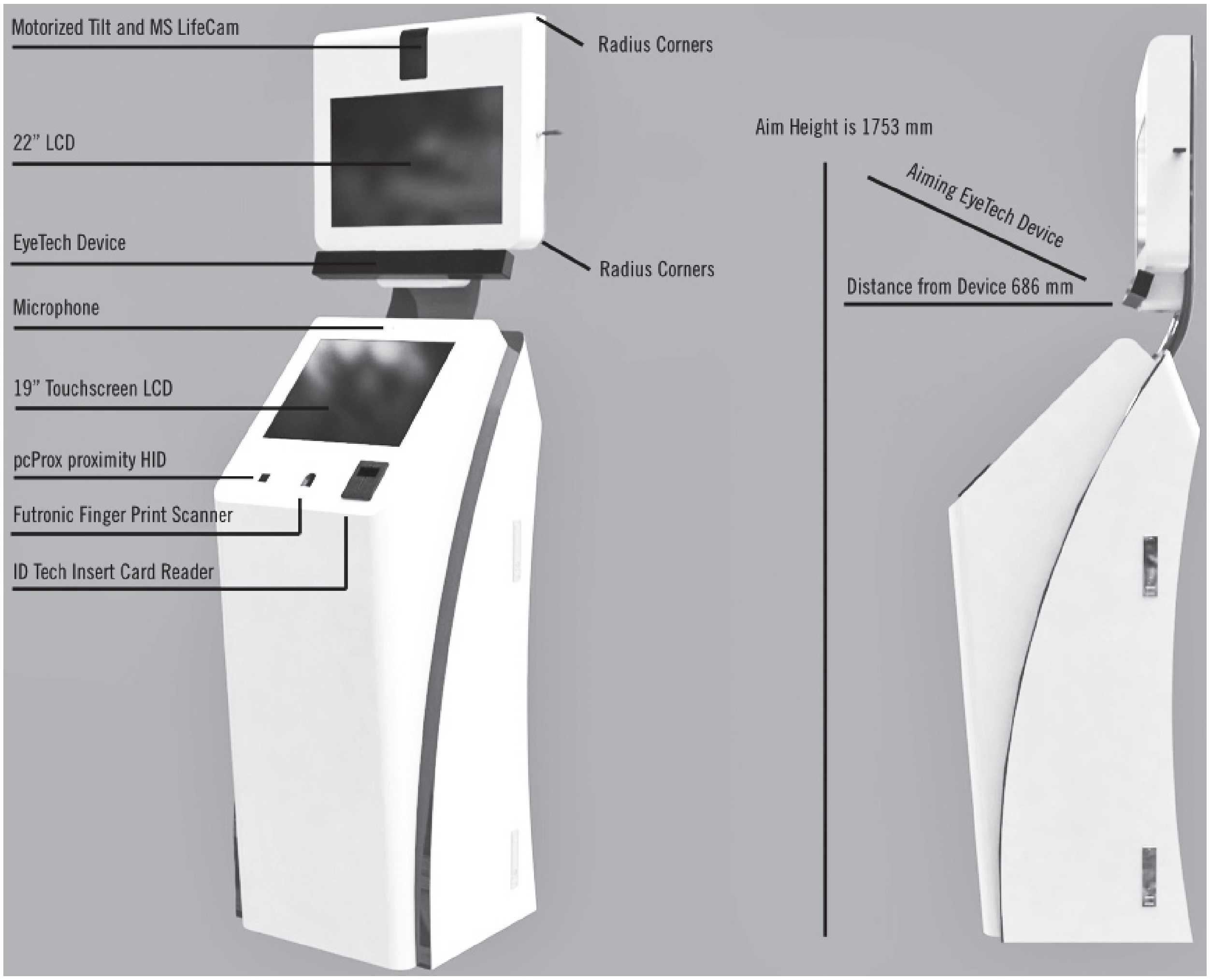

Figure 8. The first-generation credibility assessment kiosk. The kiosk contains an array of sensors for detecting indications of deception, and two video screens. On the top screen, which is approximately at eye level, the kiosk displays an animated avatar that asks scripted questions and delivers canned responses, depending on whether the system detects signals known to be associated with deception. The lower screen displays images and system controls to users.

1. Agres, A., de Vreede, G., and Briggs, R. A tale of two cities: case studies of GSS transition in two organizations. *Group Decision and Negotiation*, 14, 4 (2005), 267–284.
2. Alter, S. Work System Theory: Overview of Core Concepts, Extensions, and Challenges for the Future. *Journal of the Association for Information Systems*, 14, 2 (2013), 72-121.

3. Anson, R., Bostrom, R., and Wynne, B. An experiment assessing group support system and facilitator effects on meeting outcomes. *Management Science*, 41, 2 (1995), 189-208.
4. Bacharach, S. Organizational theories: Some criteria for evaluation. *Academy of Management Review*, 14, 4 (1989), 496-515.
5. Badura, V., Read, A., Briggs, R.O., and de Vreede, G.J. Coding for Unique Ideas and Ambiguity: A Method for Measuring the Effect of Convergence on the Artifact of an Ideation Activity. *International Journal of Social and Organizational Dynamics in IT (IJSODIT)*, 1, 3 (2011), 1-17.
6. Barrouillet, P., and Camos, V. The time-based resource-sharing model of working memory. *The cognitive neuroscience of working memory* (2007), 59–80.
7. Bhattacherjee, A. Understanding information systems continuance: An expectation-confirmation model. *Management Information Systems Quarterly*, 25, 3 (2001), 351-370.
8. Boehm, B.W., Grünbacher, P., and Briggs, R.O. EasyWinWin: A groupware-supported methodology for requirements negotiation. *Proceedings of the 23rd International Conference on Software Engineering*, Hawaii: IEEE, 1997, pp. 720-721.
9. Briggs, R., de Vreede, G., Jr., J.N., and Tobey, D. ThinkLets: Achieving Predictable, Repeatable Patterns of Group Interaction with Group Support Systems (GSS). *Proceedings of the 34th Annual Hawaii International Conference on System Sciences*, Hawaii, USA: IEEE Computer Society, 2001, pp. 1057.
10. Briggs, R., Kolfschoten, G., and de Vreede, G. Toward a theoretical model of consensus building. *Americas Conference on Information Systems AMCIS 2005*, Omaha, NE, 2005, pp. 1-12.
11. Briggs, R., Kolfschoten, G., de Vreede, G., Albrecht, C., Dean, D., and Lukosch, S. A Seven Layer Model of Collaboration: Separation of Concerns for Designers of Collaboration Systems. *30th International Conference on Information Systems*, Phoenix, AZ: Association for Information Systems, 2009, pp. 1-14 (On CD)
12. Briggs, R.O., de Vreede, G.J., and Nunamaker, J.F. Collaboration Engineering with thinkLets to Pursue Sustained Success with Group Support Systems. *Journal of Management Information Systems*, 19, 4 (2003), 31-63.
13. Briggs, R.O., Kolfschoten, G.L., de Vreede, G.-J., Lukosch, S., and Albrecht, C.C. Facilitator-in-a-box: process support applications to help practitioners realize the potential of collaboration technology. *Journal of Management Information Systems*, 29, 4 (2013), 159-194.

14. Briggs, R.O., Kolfschoten, G.L., Vreede G.J., d., Albrecht, C., Lukosch, S.G., and Dean, D.L. A Six Layer Model of Collaboration. In, Nunamaker Jr, J.F., Romano Jr, N.C., and Briggs, R.O., (eds.), *Collaboration Systems: Concept, Value, and Use*, Armonk, New York; London: M.E. Sharp, 2014, pp. 211-228.
15. Briggs, R.O., and Murphy, J.D. Discovering and Evaluating Collaboration Engineering Opportunities: An Interview Protocol Based on the Value Frequency Model. *Group Decision and Negotiation*, 20, 3 (2011), 315-346.
16. Briggs, R.O., and Randrup, N. Evaluating the Performance of Collaboration Engineers. *Proceeding of the 48th Hawaii International Conference on System Sciences*, Poipu, Kawaii, HI, 2015, pp. 1-11 (On CD).
17. Briggs, R.O., and Reinig, B.A. Bounded Ideation Theory. *Journal of Management Information Systems*, 27, 1 (2010), 123-144.
18. Briggs, R.O., Reinig, B.A., and de Vreede, G.J. The Yield Shift Theory of Satisfaction and its application to the IS/IT domain. *Journal of the Association for Information Systems*, 9, 5 (2008), 267-293.
19. Briggs, R.O., and Schwabe, G. On Expanding the Scope of Design Science in IS Research. *Proceedings of DESRIST 2011 The Sixth Design Science Research in Information Systems and Technologies Conference*, 2011, pp. 92-106.
20. Briggs, R.O., Vreede, G.J.d., and Kolfschoten, G.L. Report of the HICSS-38 Workshop on Collaboration Engineering.: Hawaii International Conference on System Sciences, 2006, pp. 1-11.
21. Connolly, T., Jessup, L.M., and Valacich, J.S. Effects of Anonymity and Evaluative Tone on Idea Generation in Computer-Mediated Groups. *Management Science*, 36, 6 (1990), 689-703.
22. Dean, D.L., Hender, J.M., Rodgers, T.L., and Santanen, E.L. Identifying quality, novel, and creative Ideas: Constructs and scales for idea evaluation. *Journal of the Association for Information Systems*, 7, 10 (2006), 646-698.
23. Dennis, A.R., Nunamaker, J.F., and Vogel, D.R. A comparison of laboratory and field research in the study of electronic meeting systems. *Journal of Management Information Systems*, 7, 3 (1990), 107-135.
24. Dennis, A.R., Valacich, J.S., Connolly, T., and Wynne, B.E. Process structuring in electronic brainstorming. *Information Systems Research*, 7, 2 (1996), 268-277.
25. Derrick, D.C., Elkins, A.C., Burgoon, J.K., Nunamaker, J.F., and Zeng, D.D. Border Security Credibility Assessments via Heterogeneous Sensor Fusion. *IEEE Intelligent Systems*, 25, 3 (2010), 41-49.
26. Diehl, M., and Stroebe, W. Productivity Loss in Brainstorming Groups: Toward the Solution of a Riddle. *Journal of Personality and Social Psychology*, 53, 3 (1987), 497-509.

27. Fjermestad, J., and Hiltz, S. Group support systems: A descriptive evaluation of case and field studies. *Journal of Management Information Systems*, 17, 3 (2000), 115-159.
28. Fjermestad, J., and Hiltz, S.R. An assessment of group support systems experimental research: methodology and results. *Journal of Management Information Systems*, 15, 3 (1998), 7-149.
29. Gallupe, R., DeSanctis, G., and Dickson, G. Computer-Based Support for Group Problem-Finding: An Experimental Investigation. *MIS quarterly*, 12, 2 (1988), 277-296.
30. Gallupe, R.B., Dennis, A.R., Cooper, W.H., Valacich, J.S, Bastianutti, L.M, & Nunamaker Jr., J.F. Electronic brainstorming and group size. *Academy of Management Journal*, 35, 2 (1992), 350-369.
31. Gamer, M., Verschuere, B., Ben-Shakhar, G., and Meijer, E. Detecting concealed information using autonomic measures. In, Verschuere, B., Ben-Shakhar, G., and Meijer, E., (eds.), *Memory detection: Theory and application of the Concealed Information Test*, Cambridge: Cambridge University Press, 2011, pp. 27-45.
32. Gregor, S. The nature of theory in information systems. *Management Information Systems Quarterly*, 30, 3 (2006), 611-642.
33. Gregor, S., and Jones, D. The anatomy of a design theory. *Journal of the Association for Information Systems*, 8, 5 (2007), 312-335.
34. Grohowski, R., McGoff, C., Vogel, D., Martz, B. Jr., & Nunamaker, J.F. Jr. Implementing Electronic Meeting Systems at IBM: Lessons Learned and Success Factors. *Management Information Systems Quarterly*, 14, 4 (1990), 368-383.
35. Gruenbacher, P. Collaborative requirements negotiation with EasyWinWin. IEEE, 2000, pp. 954-958.
36. Grünbacher, P., and Briggs, R.O. Surfacing tacit knowledge in requirements negotiation: Experiences using EasyWinWin. *Proceedings of the 34th Hawaii International Conference on System Sciences*: IEEE, 2001, pp. 1-10, On CD.
37. Hancock, J.T., Curry, L.E., Goorha, S., and Woodworth, M. On lying and being lied to: A linguistic analysis of deception in computer-mediated communication. *Discourse Processes*, 45, 1 (2007), 1-23.
38. Heinrich, P., Kilic, M., Aschoff, F.-R., and Schwabe, G. Enabling relationship building in tabletop-supported advisory settings. *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing*: ACM, 2014, pp. 171-183.

39. Hevner, A., and Chatterjee, S. Design Science Research: Looking to the Future. In, Hevner, A., and Chatterjee, S., (eds.), *Design Research in Information Systems*, New York: Springer US, 2010, pp. 261-268.
40. Jones, A.N., and Miller, D.E. Using technology for stakeholder consultation. The World Bank's use of GroupSystems V in developing countries. *System Sciences, 1997, Proceedings of the Thirtieth Hawaii International Conference on*: IEEE, 1997, pp. 464-468.
41. Kirkpatrick, D. Here Comes the Payoff From PC's. *Fortune*, March 23, 1992, pp. 93-100.
42. Kolfschoten, G.L., Briggs, R.O., De Vreede, G.-J., Jacobs, P.H., and Appelman, J.H. A conceptual foundation of the thinkLet concept for Collaboration Engineering. *International Journal of Human-Computer Studies*, 64, 7 (2006), 611-621.
43. Kolfschoten, G.L., Duivenvoorde, G., Briggs, R.O., and Vreede, G.J.d. Towards an instrument to measure successfulness of collaborative effort from a participant perspective. *Proceedings of the 42nd Hawaii International Conference on System Sciences*: IEEE, 2009, pp. 1-10, on CD.
44. Kolfschoten, G.L., Lowry, P.B., Dean, D.L., and Kamal, M. A Measurement Framework for Patterns of Collaboration. In, Briggs, R.O., Vreede G.J., d., and Kolfschoten, G.L., (eds.), *Report of the HICSS-41 Workshop on Collaboration Engineering*, http://www.hicss.hawaii.edu/Reports.htm: Hawaii International Conference on System Sciences, 2008, pp. 5-21.
45. Konsynski, B.R., Kottemann, J.E., Nunamaker, J.F., and Stott, J.W. PLEXSYS-84: an integrated development environment for information systems. *Journal of Management Information Systems*, 1, 3 (1984), 64-104.
46. Lam, S. The effects of group decision support systems and task structures on group communication and decision quality. *Journal of Management Information Systems*, 13, 4 (1997), 193-215.
47. Luna-Reyes, L.F., Derrick, D.C., Langhals, B., and Nunamaker, J.F. Collaborative Cross-Border Security Infrastructure and Systems: Identifying Policy, Managerial and Technological Challenges. *International Journal of E-Politics (IJEP)*, 4, 2 (2013), 21-38.
48. Munkvold, B.E., and Anson, R. Organizational adoption and diffusion of electronic meeting systems: a case study. *Group ’01 Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work*: ACM, 2001, pp. 279-287.
49. Nunamaker, J., Dennis, A., Valacich, J., Vogel, D., and George, J. Electronic meeting systems. *Communications of the ACM*, 34, 7 (1991), 40-61.

50. Nunamaker, J., Vogel, D., Heminger, A., Martz, B., Grohowski, R., and McGoff, C. Experience at IBM with group support systems: a field study. *Decision Support Systems*, 5, 2 (1989), 183-196.
51. Nunamaker, J.F., Applegate, L.M., and Konsynski, B.R. Computer-Aided Deliberation: Model Management and Group Decision Support Special Focus Article. *Operations Research*, 36, 6 (1988), 826-848.
52. Nunamaker, J.F., Burgoon, J.K., Twyman, N.W., Proudfoot, J.G., Schuetzler, R.M., and Giboney, J.S. Establishing a foundation for automated human credibility screening. *IEEE International Conference on Intelligence and Security Informatics*, Washington, D.C. , 2012, pp. 202-211.
53. Nunamaker, J.F.J., Briggs, R.O., Mittleman, D.D., Vogel, D., and Balthazard, P.A. Lessons from a Dozen Years of Group Support Systems Research: A Discussion of Lab and Field Findings. *Journal of Management Information Systems*, 13, 3 (1997), 163-207.
54. Nunamaker Jr, J., Applegate, L., and Konsynski, B. Facilitating group creativity: Experience with a group decision support system. *Journal of Management Information Systems*, 3, 4 (1987), 5-19.
55. Nunamaker Jr, J., Burgoon, J.K., Twyman, N.W., Proudfoot, J.G., Schuetzler, R., and Giboney, J.S. Establishing a foundation for automated human credibility screening. *2012 IEEE International Conference on Intelligence and Security Informatics (ISI),* , Arlington, VA: IEEE, 2012, pp. 202-211.
56. Nunamaker Jr., J.F., and Briggs, R.O. Toward a broader vision for Information Systems. *ACM Transactions on Management Information Systems (TMIS)*, 2, 4 (2011), 20:21-20:12.
57. Nussbaumer, P., Matter, I., and Schwabe, G. "Enforced" vs. "Causal" Transparency -- findings from IT-Supported Financial Advisory Encounters. *ACM Transactions on Management Information systems*, 3, 2 (2012), 11:11-11:19.
58. Osborn, A. *Applied imagination: Principles and procedures of creative thinking*. Scribner New York, 1953.
59. Post, B.Q. A Business Case Framework for Group Support Technology. *Journal of Management Information Systems*, 9, 3 (1993), 7-26.
60. Read, A., Hullsiek, B., and Briggs, R.O. The Seven Layer Model of Collaboration: An Exploratory Study of Process Identification and Improvement. *Proceedings of the 45th Hawaii International Conference on System Sciences*: IEEE, 2012, pp. 1-10 (On CD).

61. Reinig, B.A., Briggs, R.O., and Nunamaker, J.F., Jr. On the measurement of ideation quality. *Journal of Management Information Systems*, 23, 4 (2007), 143-162.
62. Santanen, E. Opening the black box of creativity: Causal effects in creative solution generation. In, Thompson, L.L., and Choi, H.-S., (eds.), *Creativity and innovation in organizational teams*, Mahwah, New Jersey: Lawrence Erlbaum Associates, Publishers, 2006, pp. 21-42.
63. Santanen, E.L., Briggs, R.O., and de Vreede, G.J. Causal relationships in creative problem solving: comparing facilitation interventions for ideation. *Journal of Management Information Systems*, 20, 4 (2004), 167-198.
64. Sherpherd, M.M., Briggs, R.O., Reinig, B.A., and Yen, J. Social loafing in electronic brainstorming: Invoking social comparison through technology and facilitation techniques to improve group productivity. *System Sciences, 1995. Vol. IV. Proceedings of the Twenty-Eighth Hawaii International Conference on*: IEEE, 1995, pp. 523-532.
65. Sosik, J., Avolio, B., and Kahai, S. Effects of Leadership Style and Anonymity on Group Potency and Effectiveness in a Group Decision Support System Environment. *Journal of Applied Psychology*, 82, 1 (1997), 89-103.
66. Stebbins, R. *Exploratory research in the social sciences*. Thousand Oaks, CA; London, UK; New Delhi, India: Sage Publications, Inc, 2001.
67. Teichroew, D., and Hershey III, E. PSL/PSA: A computer-aided technique for structured documentation and analysis of information processing systems. *IEEE Transactions On Software Engineering*, 3, 1 (2006), 41-48.
68. Thomson, A.M., Perry, J.L., and Miller, T.K. Conceptualizing and measuring collaboration. *Journal of Public Administration Research and Theory*, 19, 1 (2009), 23-56.
69. Vreede, G.J.d., and Briggs, R.O. ThinkLets: Five examples of creating patterns of group interaction. In, Ackerman, F., and Vreede, G.J.d., (eds.), *Proceedings of Group Decision and Negotiation, La Rochelle, France, June 4-7*, 2001, pp. 199-208.
70. Vreede, G.J.d., Briggs, R.O., and Massey, A.P. Collaboration Engineering: Foundations and Opportunities: Editorial to the Special Issue on the Journal of the Association of Information Systems. *Journal of the Association for Information Systems*, 10, 3 (2009), 121-137.
71. Yin, R.K. *Case Study Research: Designs and Methods*. Newbury park: Sage Publications, 2013.